

REMARKS

This is in response to the Office Action mailed on May 18, 2007. Claims 7-13, 15 and 25-36 were pending in the application, and the Examiner rejected all claims. With this amendment, claims 7-10, 13, 15, 25 and 33 are amended, and the remaining claims are unchanged in the application.

The Examiner rejected all pending claims under 35 U.S.C. §103 as being unpatentable over Watkins et al. (US 2004/2204953) in view of Lo (US Patent No. 5,758,347). Applicant respectfully traverses the Examiner's rejection. Of the pending claims, claims 7, 25 and 33 are independent claims.

In some prior systems that separated language-specific portions from language-neutral portions of resources, the language-specific portions of the resources, for multiple different languages, were all stored together in a language-specific resource file. Similarly, the resources that corresponded to individual, different applications were all stored separately. In other words, the language dependent resources for one application, when they were separated from the language neutral portions, were stored separately for each application. This resulted in a system architecture in which a relatively large number of language-specific resource files were generated. However, each of those resource files was associated with a different application, and each of them also had resources of multiple different languages in them.

The result was that when a plurality of different applications were loaded into a computer operating system, then a plurality of different resource files needed to be loaded as well (at least one for each application). Each of those resource files also included unwanted information. In other words, if the language being used were English, and the resource files each included language-specific portions for English, French, and Japanese, then the French and Japanese portions would be loaded as well, since they were part of the same resource file as the English portion. This further resulted in an increased number of I/O operations required to be performed by the computer system, thus degrading its performance.

The present system addresses these deficiencies in two ways. First, each compacted resource file stores language-specific resource files having language-specific resource

portions, all of a single language. For instance, a compacted resource file might store only English language specific portions of resources. Another might store only the French version, etc.

The compacted resource file stores a plurality of different language-specific resource files, but all of which are in the English language, for example. The compacted resource files for resources in the French language and Japanese language, for instance, are separate compacted resource files. In this way, when the computer system loads the language-specific resource files for the appropriate language, it need not load a large amount of unwanted, and unnecessary, data. It is only loading the resource files for the language that is actually being used.

Another feature of the present system compacts, in a single compacted resource file, language-specific resources that are used by a plurality of separate applications. For instance, where two applications are often used together (e.g., where the applications comprise different components of a single, larger, application), the resources files (all of a same language) for all of the related applications (or components) are compacted into the same compacted resource file. In this way, when the computer system loads the related applications, it need only load a single compacted resource file, and all resources for the related applications are loaded with that single compacted resource file. Again, this reduces the number of I/O operations required by the computer and enhances performance.

The first feature discussed above is set out in independent claim 7 of the present application. Independent claim 7 is drawn to a computer readable medium having instructions for creating a compacted resource file. The instructions include “reading a control file, wherein the control file specifies a compacted resource file and a plurality of language dependent resource files that are to be compacted into the compacted resource file, all of the plurality of language dependent resource files to be compacted into the compacted resource file being of a same language, and each language dependent resource file being associated with a language neutral code portion of one of a plurality of applications...”. Claim 7 further states “storing the compacted resource file header in the compacted resource file, such that the compacted resource

file includes resource portions, all having resource information of a same language, stored for access by the language neutral code portion of the plurality of different applications.” It is thus clear that independent claim 1 is directed to creating a compacted resource file where the resource information in that compacted resource file is all of the same language. This yields the benefits discussed above.

In order to meet these limitations, the Examiner cited Watkins. In particular, the Examiner cited FIG. 3 of Watkins. This shows the opposite of what is claimed by independent claim 7. It is clear from FIG. 3 that the resource file 36 contains multiple different languages (English, French, etc.) instead of only a single language. It is therefore explicitly clear that Watkins teaches the opposite of independent claim 7. While claim 7 teaches that the compacted resource file only has resources that are in the same language, FIG. 3 and the text cited by the Examiner indicate that Watkins teaches that the resource file has resources for multiple different languages. This results in increased I/O operations for the computer and degraded performance, as discussed above.

The fact that FIG. 3 of Watkins shows resource files of different languages is acknowledged by the Examiner. For instance, on page 2 of the Office Action, the Examiner stated “...clearly in FIG. 3 there is resource information, resource file index and language index with different languages all stored in one compact resource file.” Thus, the Examiner acknowledges that FIG. 3 shows just the opposite of what is claimed in independent claim 7. Applicant thus submits that independent claim 7 is allowable.

Of course, as stated in prior responses, Lo does not remedy this deficiency, and the Examiner has not asserted otherwise. Lo is simply drawn to using a storage manager to manage data in different, layered, configurations. It does not teach or suggest storing a plurality of different language dependent resource files of a same language, used by different applications, in a single compacted resource file.

Applicant thus submits that independent claim 7 is allowable over the references cited by the Examiner. Applicant further submits that dependent claim 8-13 and 15, which depend either directly or ultimately from independent claim 7 are allowable as well.

Independent claim 25 is similar to independent claim 7 in that it specifically states that all of the resources in the compacted resource file are of the same language. Claim 25 specifically states “the plurality of resource files being language dependent resource files, all of a same language...”. Claim 25 also states “storing the compacted resource file header in the compacted resource file such that the compacted resource file has resources, all of the same language...”. Therefore, claim 25 is allowable for the same reasons as claim 7.

Also, however, independent claim 25 is directed to the second advantageous feature mentioned above, in that it stores resources for related applications, in a single compacted resource file. Therefore, when the related applications are loaded, only a single compacted resource file needs to be loaded in order to accommodate the resources for those applications. This reduces the number of I/O operations required and thus enhances computer performance.

Specifically, independent claim 25 states that the plurality of resource files are “used by language neutral code portions of a plurality of different, related, applications...”. Claim 25 goes on to state that the compacted resource file is stored “such that the compacted resource file has all resources, all of the same language, for the related applications stored for access by the plurality of different, related, applications.”

This feature is neither taught nor suggested by the references cited by the Examiner either. There is no mention in the references that all resources of related applications are stored together, in a single compacted resource file, so that when one or more of the related applications are loaded, only a single compacted resource file needs to be loaded in order to accommodate the resources for all related applications. Thus, Applicant submits that independent claim 25 is allowable for this reason as well.

Independent claim 33 is similar to independent claim 7 as well, in that it specifically states that the compacted resource file has resources all of a same language. Specifically, independent claim 33 claims that the resource tool is configured to read a plurality of resource files “being language dependent resource files, all of a same language...and store the compacted resource file so the compacted resource file has resources, all of the same

language...". Therefore, Applicant submits that independent claim 33 is allowable for the same reasons as independent claim 7.

Additionally, however, independent claim 33 claims that all of the resources for different components of a single application are stored together in the same compacted resource file. In this way, if a larger application has multiple application components, each with its own set of resource files, only a single compacted resource file needs to be loaded to accommodate all the resources for the larger, multi-component, application. Specifically, independent claim 33 states that the plurality of resource files are "used by a plurality of different language neural code components of a single application...". Similarly, claim 33 specifically states that the compacted resource file has resources "stored for access by the plurality of different language neural code components of the single application." Neither of the references teach or suggest that the resources for the different components of a multi-component application are stored together in a single compacted resource file, where all of the resources have the same language. Thus, Applicant submits that independent claim 33 is allowable as well.

In conclusion, Applicant submits that independent claims 7, 25 and 33 are allowable over the references cited by the Examiner. Applicant further submits that dependent claims 8-13, 15, 26-32 and 34-36 are allowable as well. Reconsideration and allowance of claims 1-13, 15 and 25-36 are respectfully requested.

The Director is authorized to charge any fee deficiency required by this paper or credit any overpayment to Deposit Account No. 23-1123.

Respectfully submitted,

WESTMAN, CHAMPLIN & KELLY, P.A.

By: 

Joseph R. Kelly, Reg. No. 34,847

900 Second Avenue South, Suite 1400

Minneapolis, Minnesota 55402-3319

Phone: (612) 334-3222 Fax: (612) 334-3312

JRK:slg